

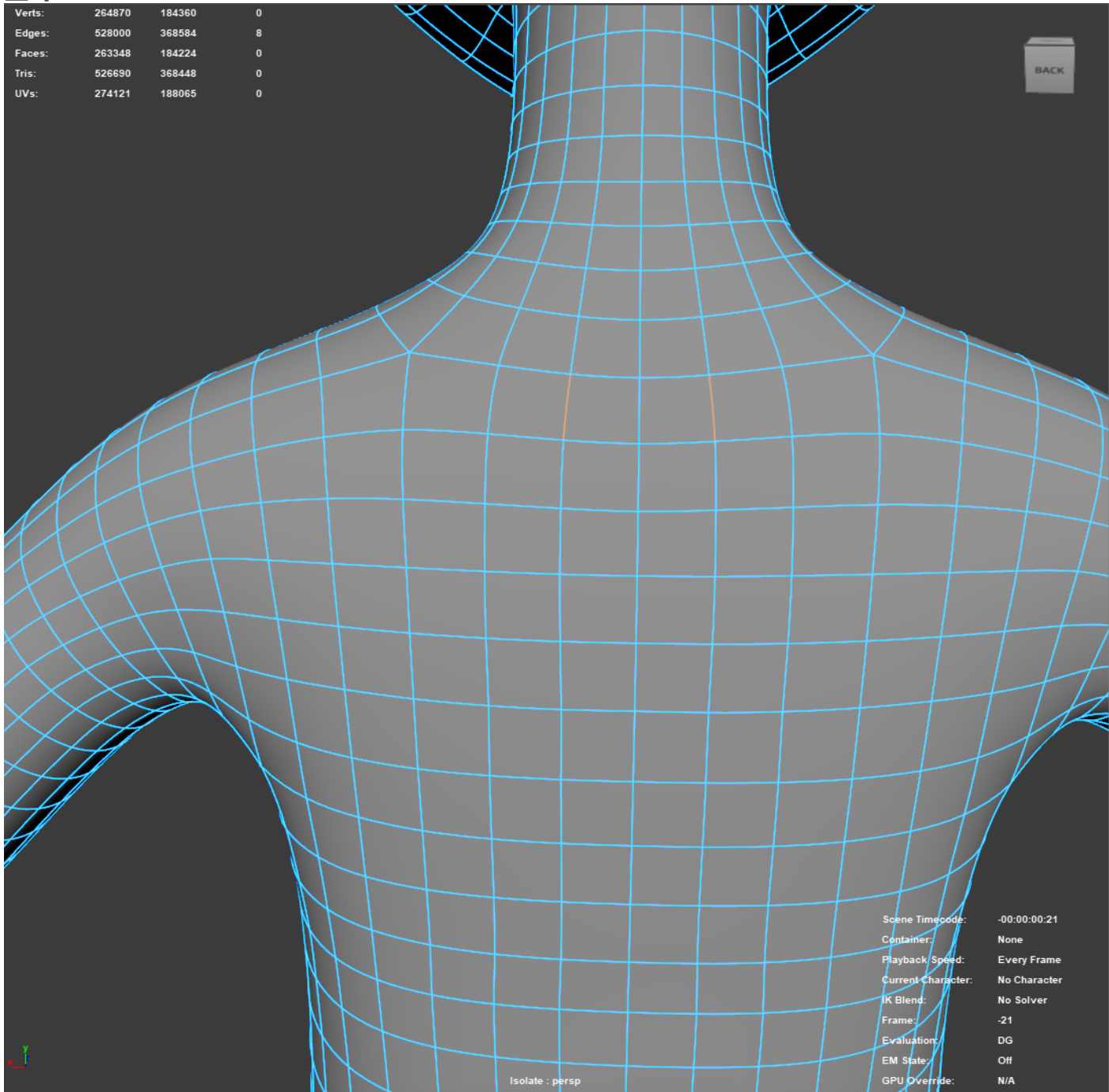
Инструкция 1.

Использование mute на nucleus

1. Выделяем два параллельных эйджа

Это два эйджа (edges) на геометрии, которые будут использоваться как база для rivet-a. Они должны быть соседними и лежать на одном участке поверхности (например, рядом на коже, ткани, волосах).

Скриншот:



2. Используем следующий скрипт для rivet

```
// Copyright (C) 2000-2001 Michael Bazhutkin - Copyright (C) 2000 studio Klassika  
// www.geocities.com/bazhutkin  
// bazhutkin@mail.ru  
//  
// Rivet (button) 1.0
```

```

// Script File
// MODIFY THIS AT YOUR OWN RISK
//
// Creation Date: April 13, 2001
//
//
// Description:
//□Use "Rivet" to constrain locator to polygon or NURBS surfaces
//□Select two edges on polygon object
//□or select one point on NURBS surface and call rivet
//□Parent your rivets and buttons to this locator

global proc string rivet ( )
{

string $nameObject;
string $namePOSI;

string $parts[];
string $list[] = `filterExpand -sm 32`;
int $size = size($list);
if ($size > 0)
{
□if ($size != 2)
□{□error("No two edges selected");
□□return "";
□}

□tokenize($list[0], ".", $parts);
□$nameObject = $parts[0];
□tokenize($list[0], "[", $parts);
□float $e1 = $parts[1];
□tokenize($list[1], "[", $parts);
□float $e2 = $parts[1];

□string $nameCFME1 = `createNode curveFromMeshEdge -n "rivetCurveFromMeshEdge1"`;
□□setAttr ".ihi" 1;
□□setAttr ".ei[0]" $e1;
□string $nameCFME2 = `createNode curveFromMeshEdge -n "rivetCurveFromMeshEdge2"`;
□□setAttr ".ihi" 1;

```

```

[]setAttr ".ei[0]" $e2;
[]string $nameLoft = `createNode loft -n "rivetLoft1"`;
[]setAttr -s 2 ".ic";
[]setAttr ".u" yes;
[]setAttr ".rsn" yes;

[]$namePOSI = `createNode pointOnSurfaceInfo -n "rivetPointOnSurfaceInfo1"`;
[]setAttr ".turnOnPercentage" 1;
[]setAttr ".parameterU" 0.5;
[]setAttr ".parameterV" 0.5;

[]connectAttr -f ($nameLoft + ".os") ($namePOSI + ".is");
[]connectAttr ($nameCFME1 + ".oc") ($nameLoft + ".ic[0]");
[]connectAttr ($nameCFME2 + ".oc") ($nameLoft + ".ic[1]");
[]connectAttr ($nameObject + ".w") ($nameCFME1 + ".im");
[]connectAttr ($nameObject + ".w") ($nameCFME2 + ".im");
}
else
{
[]$list = `filterExpand -sm 41`;
[]$size = size($list);

[]if ($size > 0)
[]{
[]if ($size != 1)
[]{[]error("No one point selected");
[]return "";
[]}
[]tokenize($list[0], ".", $parts);
[]$nameObject = $parts[0];
[]tokenize($list[0], "[", $parts);
[]float $u = $parts[1];
[]float $v = $parts[2];
[]$namePOSI = `createNode pointOnSurfaceInfo -n "rivetPointOnSurfaceInfo1"`;
[][]setAttr ".turnOnPercentage" 0;
[][]setAttr ".parameterU" $u;
[][]setAttr ".parameterV" $v;
[]connectAttr -f ($nameObject + ".ws") ($namePOSI + ".is");
[]}
[]else
[]{[]error("No edges or point selected");

```

```

[]return "";
[]
}

string $nameLocator = `createNode transform -n "rivet1"`;
createNode locator -n ($nameLocator + "Shape") -p $nameLocator;

string $nameAC = `createNode aimConstraint -p $nameLocator -n ($nameLocator + "_rivetAimConstraint1")`;
[]setAttr ".tg[0].tw" 1;
[]setAttr ".a" -type "double3" 0 1 0;
[]setAttr ".u" -type "double3" 0 0 1;
[]setAttr -k off ".v";
[]setAttr -k off ".tx";
[]setAttr -k off ".ty";
[]setAttr -k off ".tz";
[]setAttr -k off ".rx";
[]setAttr -k off ".ry";
[]setAttr -k off ".rz";
[]setAttr -k off ".sx";
[]setAttr -k off ".sy";
[]setAttr -k off ".sz";

connectAttr ($namePOSI + ".position") ($nameLocator + ".translate");
connectAttr ($namePOSI + ".n") ($nameAC + ".tg[0].tt");
connectAttr ($namePOSI + ".tv") ($nameAC + ".wu");
connectAttr ($nameAC + ".crx") ($nameLocator + ".rx");
connectAttr ($nameAC + ".cry") ($nameLocator + ".ry");
connectAttr ($nameAC + ".crz") ($nameLocator + ".rz");

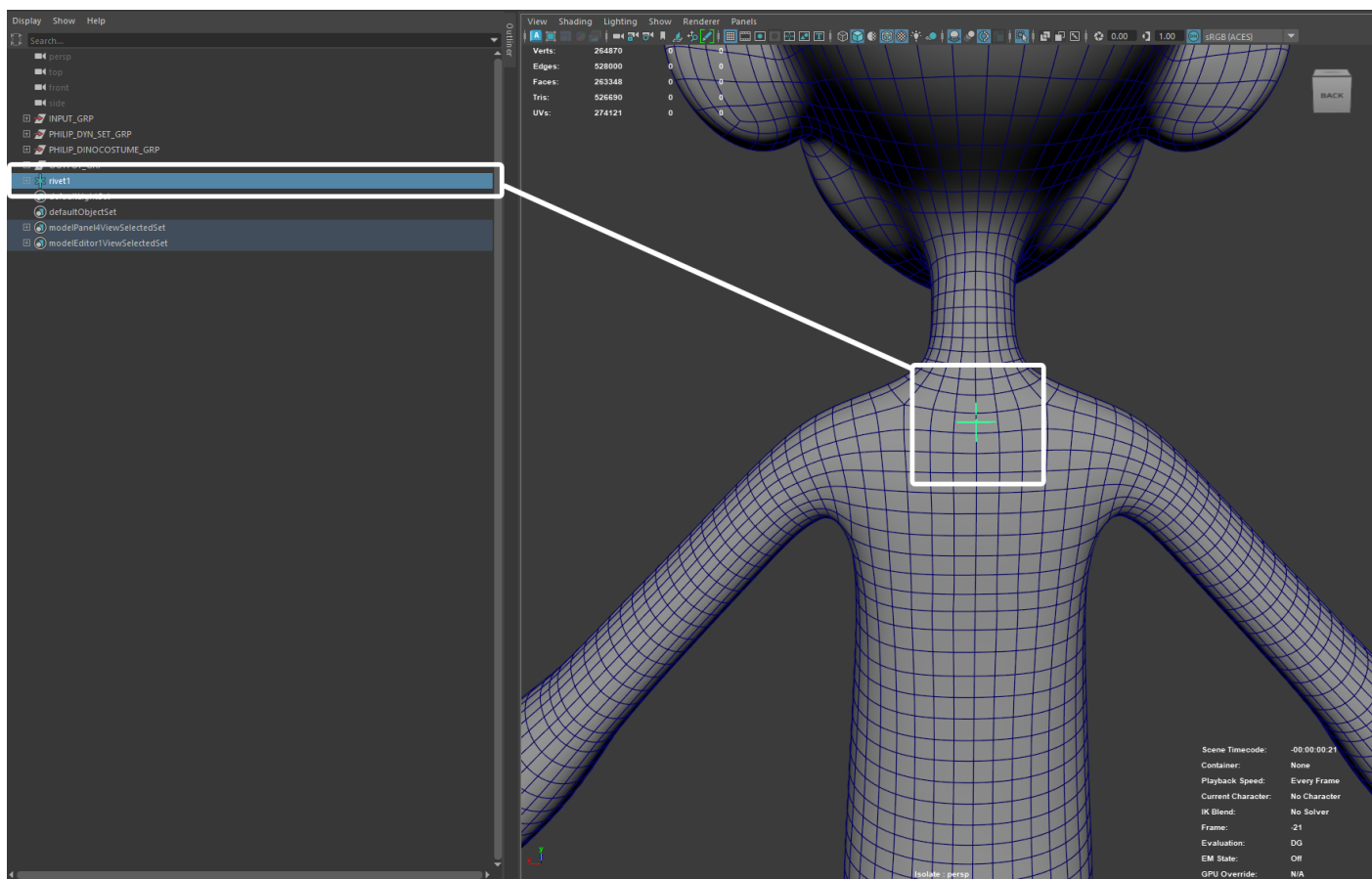
select -r $nameLocator;
return ($nameLocator);

}

rivet;

```

 **Скриншот:**



3. Переименовываем rivet в Translator

Чтобы в будущем использовать его как драйвер трансформа, нужно дать понятное имя:

`rivet1` → `Translator` (или `rivet_translate_ctrl`, `rivet_driver`, и т.п.)

Скриншот:

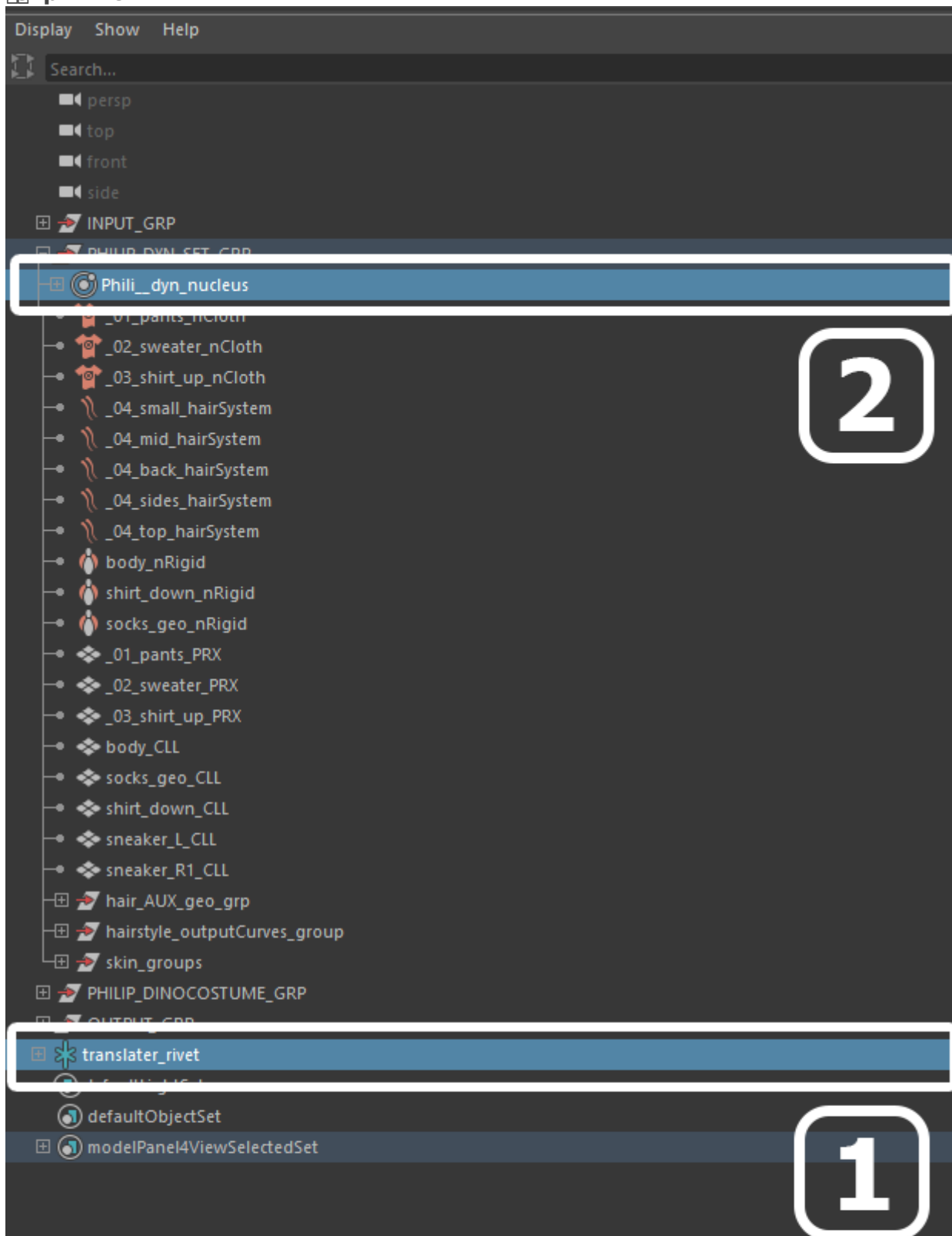


4. Выделяем Translator_rivet и объект

Сначала выбираем rivet ,к которому должен привязаться объект.

А потом геометрию или объект который мы хотим перемещать.

Скриншот:



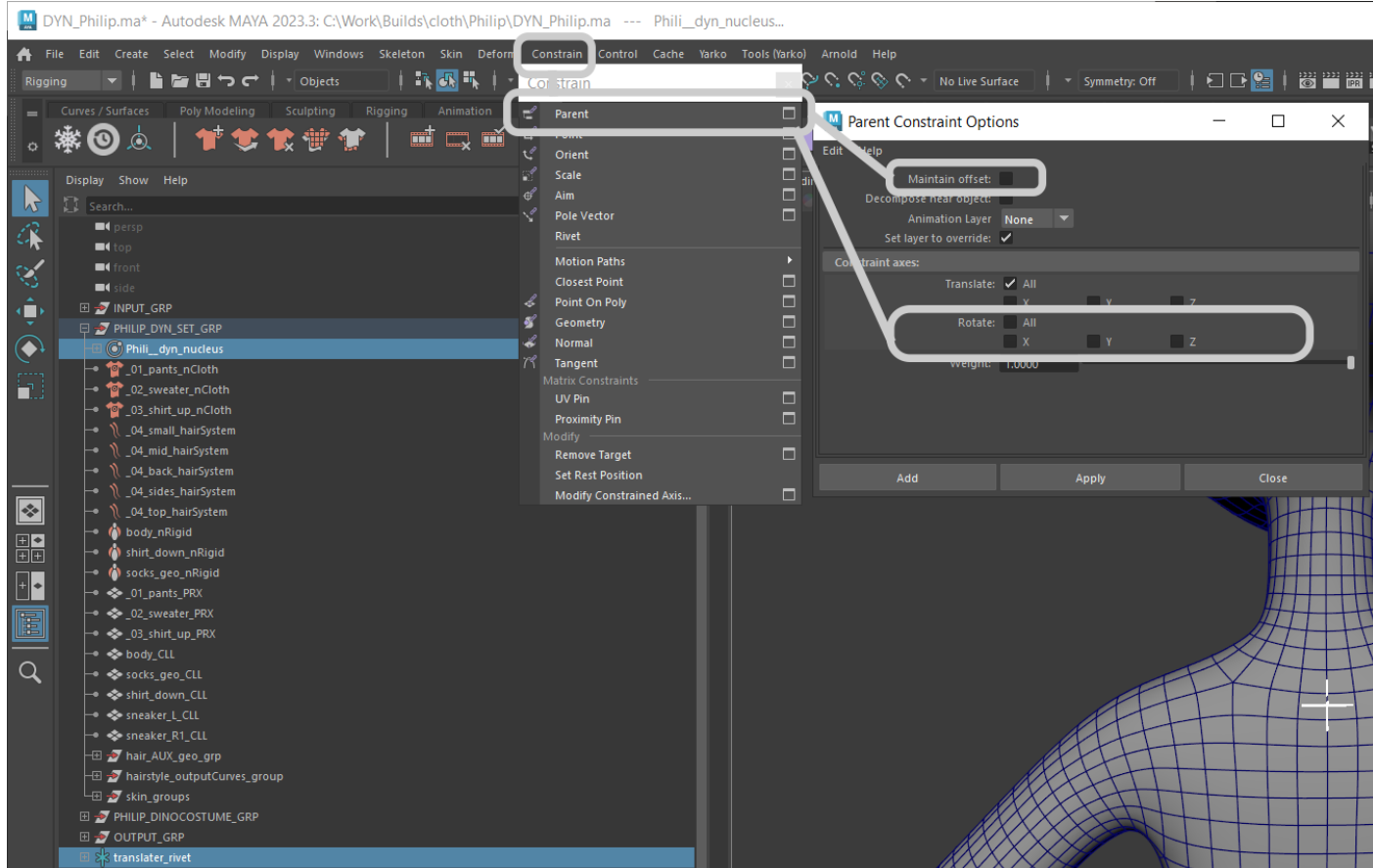
5. Выбираем Constraint → Parent

Rigging меню → Constraint → Parent Constraint

Открываем окно опций:

Rigging > Constraint > Parent (с коробочкой справа от названия)

Скриншот:



Очень важно!

Снимаем ☒ Maintain Offset, чтобы объект точно следовал движению Translater, а не сохранял своё смещённое положение.

Жмём **apply**

6. Запускаем симуляцию

Всё готово! Теперь `Translator` (rivet) двигается вместе с поверхностью, а привязанный объект точно следует за ним — можно запускать симуляцию.

Версия #5

Guest создал 24 марта 2025 18:01:36

Дмитрий Гузеев обновил 27 марта 2025 15:19:15